Aufgabe 1

Beweisen Sie die Formel

$$\{true\} \quad x := 7; y := x + 3 \quad \{y = 10\}$$

im Hoare-Kalkül.
Idee von Tobi:
kurze Regelkunde:

$$\{P[x \leftarrow e]\} \quad x := e \quad \{P\} \qquad \text{(Zuweisung)}$$

Mit anderen Worten: Man ersetzt in der Nachbedingung alle vorkommen von x durch e.
and here we go: (von hinten nach vorne, also von unten nach oben lesen ;))

$$\begin{array}{lll}
\{true\} & \{true\} & \\
\{true\} & \{7 + 3 = 10\} & \text{(Zuweisung)} \\
\{true\} \quad x := 7 & \{x + 3 = 10\} & \text{(Zuweisung)} \\
\{true\} \quad x := 7; y := x + 3 & \{y = 10\} &
\end{array}$$

Struktex Version (Hinnerk):

$\{P\} true$
$7 = 7 \Rightarrow true$
$x := 7$
$x + 3 = 10 \Rightarrow x = 7$
$y := x + 3$
$\{Q\} y = 10$

Aufgabe 2

Schreiben Sie ein WHILE'-Programm zur Berechnung der Signum-Funktion und beweisen Sie seine Korrektheit im Hoare-Kalkül.
Idee von Tobi:
Erstmal die Regeln

$$\frac{\{I \wedge B\} \quad S \quad \{I\}}{\{I\} \quad \underline{while}\ B\ \underline{do}\ S \quad \{I \wedge \neg B\}} \qquad \text{(while)}$$

```
sum:=0;
while not eof do
  read x;
  sum := sum + x;
output sum
```

Die Vorbedingung P ist relativ egal. Viel wichtiger ist die Nachbedingung Q, die muss den Zustand $in = \epsilon$ enthalten, da das Programm terminieren muss und der Wahrheitswert von $\overline{eof}$ davon abhängt. Im folgenden werden die Variablen out für die Ausgabe und in für die Eingabe verwendet. Das Symbol ϵ stellt hier eine leere Menge dar.
Die Invariante für die Whileschleife sollte eventuell nur Festhalten, dass die Ausgabe unverändert bleibt, die Schleifenbedingung lässt sich wie folgt definieren:

$B = \{\neg(in = \epsilon)\} = \{in \neq \epsilon\} = \{in = [n_0, \ldots, n_i]\}$

$$\begin{array}{lll}
 & P = \{True\} & \\
\{\} & sum := 0; & \{out = \epsilon \wedge in \neq \epsilon \wedge 0 + (n_0) \neq \epsilon\} \\
\{\} & \underline{while}\ \neg\underline{eof}\ \underline{do} & \{out = \epsilon \wedge in \neq \epsilon \wedge sum + (n_0) \neq \epsilon\} \\
\{\} & \quad \underline{read}\ x; & \{out = \epsilon \wedge in = [n_1, \ldots, n_i] \wedge sum + (n_0) \neq \epsilon\} \\
\{\} & \quad sum := sum + x; & \{out = \epsilon \wedge in \neq \epsilon \wedge sum + x \neq \epsilon\} \\
\{\} & \underline{output}\ sum & \{out = \epsilon \wedge in = \epsilon \wedge sum \neq \epsilon\} \\
 & Q = \{out = sum \wedge in = \epsilon\} &
\end{array}$$

Struktex Version (Hinnerk):

$\{I\} x \in \mathbb{Z}$

read x

$\{I\} x \in \mathbb{Z} \wedge (0 = 1 \vee 0 = -1 \vee 0 = 0)$

signum=0

$\{I\} x \in \mathbb{Z} \wedge (signum = 1 \vee signum = -1 \vee signum = 0)$

true	false
$\{I\} x \in \mathbb{Z} \wedge x > 0\ true$ $\{I\}(1 = 1) \vee (1 = -1) \vee (1 = 0) \Rightarrow true \vee false \vee false$ signum=1 $\{I\} x \in \mathbb{Z} \wedge (signum = 1 \vee signum = -1 \vee signum = 0)$ $\varnothing$	$\{I\} x \in \mathbb{Z} \wedge x \leq 0 \wedge (signum = 1 \vee signum = -1 \vee signum = 0)$ skip $\{I\} x \in \mathbb{Z} \wedge (signum = 1 \vee signum = -1 \vee signum = 0)$

$\{I\} x \in \mathbb{Z} \wedge (signum = 1 \vee signum = -1 \vee signum = 0)$

true	false
$\{I\} x \in \mathbb{Z} \wedge x < 0\ true$ $\{I\}(-1 = 1) \vee (-1 = -1) \vee (-1 = 0) \Rightarrow false \vee true \vee false$ signum=-1 $\{I\} x \in \mathbb{Z} \wedge (signum = 1 \vee signum = -1 \vee signum = 0)$ $\varnothing$	$\{I\} x \in \mathbb{Z} \wedge x \geq 0 \wedge (signum = 1 \vee signum = -1 \vee signum = 0)$ skip $\{I\} x \in \mathbb{Z} \wedge (signum = 1 \vee signum = -1 \vee signum = 0)$

$\{I\} x \in \mathbb{Z} \wedge (signum = 1 \vee signum = -1 \vee signum = 0)$

Aufgabe 3

Führen Sie einen Korrektheitsbeweis unter Verwendung der axiomatischen Semantik zu folgendem Programm:

```
sum:=0;
while not eof do
  read x;
  sum := sum + x;
output sum
```

Hinnerks Idee:

$\{I\} sum \in \mathbb{Z}$

sum:=0

$\{I\} sum \in \mathbb{Z}$

$\neg eof$

$\{I\} sum \in \mathbb{Z} \wedge \neg eof$

read x

$\{I\} sum \in \mathbb{Z}$

sum:=sum+x

$\{I\} sum \in \mathbb{Z}$

$\{I \wedge \neg B\} sum \in \mathbb{Z} \wedge eof$

output sum

$\{I\} sum \in \mathbb{Z}$

Aufgabe 4

Beweisen Sie die Gültigkeit des Axioms (A.4), d.h. zeigen Sie die Gültigkeit der Formel:

$$\{Q[output.T/output]\} \quad output T \quad \{Q\}$$

Wir unterscheiden zwei Fälle:

a) **Fehler:**

$\{Q[output.T/output]\}$

$\{Q[output.T/output]\} Fehler$

output T

$\{Q\} Fehler$

b) **kein Fehler:**

$\{Q[output.T/output]\}$

$\{Q[output.T/output]\}(n, (s, e', a))$

output T

$\{Q\}(s, e', a.n)$